

```

*****
* This program will calculate the recodes for the public use file. 1998 METRO
*
* Program: pufrcd.sas
* Programmed by: Diane Schwartz
* Date: March 11, 1999 - removed national specific public use file
*           recodes in order to run for both ms and national
* Revised: April 12, 1999 - corrected problem with nummor found in national
*           but applicable to ms also.
*
* Input Library: /natdata (or msdata)/(survey year)/topcodes
*                 /natdata (or msdata)/(survey year)/bridge
*                 /natdata (or msdata)/natmas (or msmas)
*
* Output Library: /natadata (or msdata)/(survey year)/recodes
*
* The following recodes are coded in this program:
*     per
*     zadult
*     zinc
*     zinc2
*     phone
*     kitchen
*     plumb
*     rooms
*     confee
*     zadeq
*     zmshc
*     amtx
*     nummor
*     histry
*
*****

```

```

/* ** set up lib names ** */
libname rcd  "$RECD";
libname bridg "$BRIDGE";
libname mast "$MASTR";
libname topc "$TOPCD";

missing D R B X M N U _;

/* ** sort master file on mscode90 and ctrlnm13 to match the data sets sort */

proc sort data=mast.master out=master;
  by mscode90 ctrlnm13;

```

```

run;

/* ** create a view to process the recodes */
data viewed1 / view=viewed1;
  merge topc.houshld (in = hhldend
    keep = ctrlnm13 amte amtf amtg amti amto amtt amtw
      amtxq amtxre badstep bedrms bigp burner busin
        camfq climb confeeq cook cracks dens dining elev
          famrm flrent fmhotf freeze frent hequip prent
            holes hotpip ileak kexclu kitch leak living
              lrent ltsok m3rod mhotfe mice nowire numblow
                numcold numltl nunit2 othfn oven plugs
                  railok rats recrm refr rent sharpf sink
                    status tenure toilet tub txre vacancy
                      vother curper buye cstmnt mscode90 baths hel
                        regmor henum telhh telav)
  bridg.houshldr (keep = mscode90 ctrlnm13 phonenum)
  topc.person (in = endper
    keep = ctrlnm13 age hhmem rel sal mscode90);
    by mscode90 ctrlnm13;
    if not (endper and hhldend) then
      do;
        if endper then
          do;
/* put 'no houshld obs for persons ctrlnm13 = ' ctrlnm13; */
        cnohh+1;
        end;
        else
          do;
/* put 'no persons obs for houshld ctrlnm13 = ' ctrlnm13; */
        cnopr+1;
        end;
      end;
    run;

```

```

data viewed2 / view=viewed2;
  merge master (in=endmst
    keep = ctrlnm13
      convdate mscode90)
    viewed1 (in=endvw);
    by mscode90 ctrlnm13;
    if not (endmst and endvw) then
      do;

```

```
if endvw then put 'no master obs for viewedt ctrlnm13 = ' ctrlnm13;
else delete;
end;
run;
```

```
data viewedt / view = viewedt;
merge topc.mortg (in=mtgend
                  keep = ctrlnm13 amtm amtm2 hepmt1 hepmt2 hepmt3
                         inspmt inpmt2 pmt pmt2 pmt3 pmt4 ramapp
                         taxpmt txpmt2 mcnt mscode90)
                  viewed2 (in=endvwd);
by mscode90 ctrlnm13;
if not (mtgend and endvwd) then
do;
  if mtgend then put 'no viewedt obs for mortg ctrlnm13 = ' ctrlnm13;
end;
run;
```

```
/* ** this section creates a data set with the recodes included. ** */
data rcd.pf1rcd (keep = ctrlnm13 per zadult zinc zinc2 kitchen plumb
                  rooms confee zadeq zsmhc amtx nummor
                  histry mscode90 phone);

length ctrlnm13 $13 per 3. zadult 3. zinc 5. zinc2 5. kitchen $1
      plumb $1 rooms 3. confee 3. zadeq $1 zsmhc 4. amtx 4. nummor 4.
      histry $1 srvyr $4 zsmhcr 4. phone $1;
```

```
set viewedt end=send;
by mscode90 ctrlnm13;
```

```
retain incnt 0;
retain ctrlno 0;
retain ctllst 0;
retain outcnt 0;
retain per 0;
retain zadult 0;
retain zinc 0;
retain zinc2 0;
```

```
incnt+1;
/*  if(mod(incnt,500) = 0) then put 'processing obs number ' incnt; */
if first.ctrlnm13 then ctrlno+1;
```

```

if last.ctrlnm13 then ctllst+1;

if first.ctrlnm13 then
do;
per = .B;
zadult = .B;
zinc = .B;
zinc2 = .B;
rooms = .B;
confee = .B;
zsmhc = .B;
zsmhcr = .B;
amtx = .B;
nummor = .B;
ran = .B;
rac = .B;
plumb = 'B';
zadeq = 'B';
kitchen = 'B';
phone = 'B';
end;

if status = '1' then
do;

if first.ctrlnm13 then
do;
per = 0;
zadult = 0;
zinc = 0;
zinc2 = 0;
i = 1;
if (-999998 <= vother <= 999998) then
do;
zinc+vother;
zinc2+vother;
end;
end;
end;

/* per and zadult recodes - number of persons and number of adults over 18 */

if hhmem='1' then
do;
per+1;

```

```

if (age >= 18) then
do;
    zadult+1;
/*      if ctlst < 10 then
        put 'zadult = ' zadult ' and age = ' age; */
end;

/* zinc recode - income of reference person and household
members related to reference person */

if (age >= 14 and (rel = 1 or rel = 2 or rel = 20 or (rel >= 22 and
rel <= 26))) then
zinc+sal;

/* zinc2 recode - household income */

if (age >= 14) then
zinc2+sal;

end; /* end of hhmem check */

end; /* end of status check */

if last.ctrlnm13 then
do;

if status > '0' and status < '4' then
do;

/* phone recode */
if (status = '1') then do;
    if (phonenum ne '' or telhh = 1 or telav = 1) then phone = '1';
    else if (telhh = 2 or telav = 2) then phone = '2';
    else phone = 'D';
end;

/* kitchen recode - complete kitchen facilities */
kitchen = '';

if(((nunit2 = '1' or nunit2 = '2' or nunit2 = '4')
and (sink = '1')
and (refr = '1')
and (cook = '1' or burner = '1' or oven = '1'))
or
((nunit2 = '3' or nunit2 = '5'))
```

```

and (kexclu = '1')
and (sink = '1')
and (refr = '1')
and (cook = '1' or burner = '1' or oven = '1')) then
do;
    kitchen = '1';
end;
else
do;
    if ((sink = '2')
        or (refr = '2')
        or (cook = '2' and burner = '2' and oven = '2')
        or ((nunit2 = '3' or nunit2 = '5') and (kexclu = '2'))) then
        do;
            kitchen = '2';
        end;
    end;
/* plumb recode - exclusive use of plumbing */
plumb = '';

if ( hotpip = '1' and toilet = '1' and tub = '1' and sharpf = '2') then
do;
    plumb = '1';
end;
if ( hotpip = '2' or toilet = '2' or tub = '2' or sharpf = '1')then
do;
    plumb = '2';
end;

/* rooms recode - number of rooms */

rooms = 0;
rooms = sum(bedrms,kitch,living,dining,famrm,recrm,dens,busin,othfn);

if rooms >= 21 then rooms = 21;

/* confee recode - condominium or homeowners assoc. or mobile home monthly
fees */

if confeeq = . then confee = .;
if confeeq = 0 then confee = 0;
if confeeq >= 1 then
do;
    confee = (confeeq * camfq) / 12;

```

```
confeer = confee - 5;  
confee = round(confeer,10);  
confee+5;  
end;  
  
end; /* end of status check */
```

```
/* zadeq recode - adequacy of housing */  
if status = '1' then  
do;  
  zadeq = '';  
  
if (baths < 2 and (hotpip = '2'  
  or tub = '2'  
  or toilet = '2'  
  or sharpf = '1'))  
or  
  (freeze = '1'  
  and (numcold >= '3' and numcold <= '8'))  
or  
  (buye = '1')  
or  
  (nowire = '2'  
  and plugs = '2'  
  and (numblow >= '3' and numblow <= '8')) then  
do;  
  zadeq = '3';  
end;  
  
n = 0;  
if leak = '1' then n+1;  
if ileak = '1' then n+1;  
if holes = '1' then n+1;  
if cracks = '1' then n+1;  
if bigp = '1' then n+1;  
if rats = 'X' then n+1;  
if n >= 5 then  
  zadeq = '3';  
  
m = 0;  
if ltsok = '4' or ltsok = '5' then m+1;  
if badstep = '2' then m+1;
```

```

if railok = '1' or railok = '3' then m+1;
if climb >= 3 and elev ne '2' then m+1;
if m = 4 then
  zadeq = '3';

if zadeq ne '3' then
do;
  if numltl >= '3' and numltl <= '8' then
    zadeq = '2';
  if hequip = 7 then
    zadeq = '2';

n=0;
if leak = '1' then n+1;
if ileak = '1' then n+1;
if holes = '1' then n+1;
if cracks = '1' then n+1;
if bigp = '1' then n+1;
if rats = 'X' then n+1;
if n = 3 or n = 4 then
  zadeq = '2';

m=0;
if ltsok = '4' or ltsok = '5' then m+1;
if badstep = '2' then m+1;
if railok = '1' or railok = '3' then m+1;
if climb >= 3 and elev ne '2' then m+1;
if m = 3 then
  zadeq = '2';

if kitchen = '2' then
  zadeq = '2';
end;

if zadeq ne '3' and zadeq ne '2' then
  zadeq = '1';
end; /* end of status check for zadeq universe */

```

/* zsmhc - monthly housing costs recode */

```

if amtxre < 0 then amtxre = 0;
if mhotfe < 0 then mhotfe = 0;
if cstmnt < 0 then cstmnt = 0;

```

```

if status = '1' then

```

```

do;

zsmhcr = 0;
zsmhc = 0;

if (1 <= amte <= 998) then zsmhcr+amte;
if (1 <= amtg <= 998) then zsmhcr+amtg;
if (amto >= 4) then zsmhcr+(amto / 12);
if (amtf >= 4) then zsmhcr+(amtf / 12);
if (amtt >= 4) then zsmhcr+(amtt / 12);
if (amtw >= 4) then zsmhcr+(amtww / 12);

if ((amtxq >= 0) and (taxpmt ne '1' and txpmt2 ne '1')) then
do;
  zsmhcr+((amtxq - amtxre) / 12);
end;

if (amti >= 1) and (inspmt ne '1' and inpmt2 ne '1') then
do;
  zsmhcr+(amti / 12);
end;

if (confeeq >= 1) then
  zsmhcr+((confeeq*camfq)/12);
if ( 1 <= lrent <= 1996) then
  zsmhcr+((lrent * flrent) / 12);
if (nunit2 = '4' or nunit2 = '5') then
do;
  if (mhotfe >= 1) then
    zsmhcr+((mhotfe * fmhotf) / 12);
end;
if ((rent >= 2) and (rent < 1 or rent > 9997)) then
  zsmhcr+((rent*frent)/12);
if (rent >= 1 and frent >= 1) then
do;
  zsmhcr+((rent*frent) / 12);
end;
if (pmt >= 1) then
  zsmhcr+pmt;
if (pmt2 >= 1) then
  zsmhcr+pmt2;
if (pmt3 >= 1) then
  zsmhcr+pmt3;
if (pmt4 >= 1) then
  zsmhcr+pmt4;
/* if (hepmt1 >= 1) then

```

```

zsmhcr+hepmt1; */
/* if (hepmt2 >= 1) then
   zsmhcr+hepmt2; */
/* if (hepmt3 >= 1) then
   zsmhcr+hepmt3; */
if (amtm >= 1) then
  zsmhcr = zsmhcr-(amtm / 12);
if (amtm2 >= 1) then
  zsmhcr = zsmhcr-(amtm2 / 12);
if tenure = '1' then
do;
  if ((cstmnt / 12) >= 1) then
    zsmhcr+(cstmnt / 12);
end;

zsmhc = round(zsmhcr,1);
end; /* end of universe check for zsmhc */

/* amtx recode - yearly real estate taxes */

amtix = .;
if((status = '1' and tenure = '1') or ((status = '2' or status = '3') and
(vacancy = 3 or vacancy = 5))) then
do;
  if ((status = '1' and tenure = '1') and txre = '2' and amtxq = 0) then
    amtx = 0;
  else
    if((status = '1' and tenure = '1') and txre = '1' and
((amtxq - amtxre) <= 0)) then
      amtx = 0;
    else
      if((status = '2' or status = '3') and (vacancy = 3 or vacancy = 5)
and amtxq = 0) then
        amtx = 0;
      else
        if((status = '1' and tenure = '1') and txre = '2' and amtxq >= 1)
then
          amtx = amtxq;
        else
          if((status = '1' and tenure = '1') and txre = '1' and
((amtxq - amtxre) >= 1)) then
            amtx = amtxq - amtxre;
          else
            if((status = '2' or status = '3') and (vacancy = 3 or
vacancy = 5) and amtxq >= 1) then
              amtx = amtxq;

```

```

if amtx >= 1 then
do;
    amtxr = amtx - 50;
    amtx = round(amtxr,100);
    amtx+50;
end;

end; /* end of universe check for amtx */

/* nummor recode - number of mortgages */

if (status = '1' and tenure = '1') then
do;
    nummor = .;
    hcmt = .;

    if ((ramapp = '2' or ramapp = 'D' or ramapp = 'R') and
        mcnt = 0 and hel = '2')then
        nummor = 0;
    else
        do;
            if (ramapp = '1') then
                nummor = 1;
            else
                do;
                    if hel = '2' then
                        hcmt = 0;
                    if hel = '1' and (1 <= henum <= 100) then
                        hcmt = henum;
                    if (0 <= mcnt <= 7) or (0 <= hcmt <= 100) then
                        do;
                            nummor = mcnt + hcmt;
                            if (nummor >= 7) then
                                nummor = 7;
                        end;
                    end;
                end;
            if nummor < 0 and ((hel = 'D' or hel = 'R' or hel = ' ') or
                (henum = .D or henum = .R or henum = .)) then
                nummor = .D;

end; /* end of universe check for nummor */

```

```

/* recode for histry - set to 8 for adds and extras, otherwise set to a 1 */

srvyr = sysget('YRSURV');

if srvyr = '1998' then
  histry = '7';

/* finish out the top and bottom code variables: zadult, zinc, zinc2 and
then output */

  if zadult >= 11 then zadult = 11;
  if (zinc <= -10000 and zinc ne .B) then zinc = -10000;
  if zinc >= 999996 then zinc = 999996;
  if (zinc2 <= -10000 and zinc2 ne .B) then zinc2 = -10000;
  if zinc2 >= 999996 then zinc2 = 999996;

  output rcd.pflrcd;
  outcnt+1;

end;      /* end of last.ctrlnm13 processing */

if send then
do;
  put cnohh '= number of persons obs without corresponding houshld obs';
  put cnopr '= number of houshld obs without corresponding persons obs';
  put ctrlno '= number of first control numbers processed';
  put ctllst '= number of last control numbers processed';
  put incnt '= number of view obs input';
  put outent '= number of obs output';
end;

/* end of recode program */
run;

```